Final Report (98/2/1 - 99/1/31)

Grant No. F49620-98-1-0244

"Parallel Workstation Cluster for Algorithm Development, Parameter Scoping, and Data Analysis"

Submitted to

AIR FORCE OFFICE OF SCIENTIC RESEARCH
Bolling AFB, DC 20332-8050

University of Washington
Aerospace & Energetics Research Program
Box 352250
Seattle, Washington 98195-2250

Dr. Uri Shumlak
Principal Investigator

99/4/26

AFRL-SR-BL-TR-99-

0134

REPORT DOCUMENTATION PAGE

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE A...
	99/4/26	Final Technical Report 98/2/1 - 99/1/31

4. TITLE AND SUBTITLE

"Parallel Workstation Cluster for Algorithm Development, Parameter Scoping, and Data Analysis"

5. FUNDING NUMBERS

F49620-98-1-0244

6. AUTHOR(S)

SHUMLAK, Uri

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

University of Washington
Grant & Contract Services
3935 University Way NE
Seattle, WA 98105-6613

8. PERFORMING ORGANIZATION REPORT NUMBER

62-0287

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

AFOSR/NM
110 Duncan Avenue, Room B115
Bolling AFB, DC 20332-8050

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT

unclassified/unlimited

Approved for Public Release; Distribution Unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 Words)*

A cluster of workstations is connected with a fast network and used as a local parallel computer. This approach mimics the interprocessor communication and scalability of the large machines at a fraction of the cost. This system uses off-the-shelf hardware and public domain software, which makes the system simple to assemble and maintain. The parallel workstation cluster has already significantly contributed to our AFOSR project. The cluster accelerates the algorithm development process because it eliminates the delays associated with remote computing on shared resources. The parallel workstation cluster performs simulations to scope a parameter space for optimization and design. A parallel supercomputer is then used for the detailed scoping using fewer runs. This approach has helped to alleviate the over-subscription of the shared resources and has allowed us to obtain the computational results in less time.

14. SUBJECT TERMS

parallel computer, workstation cluster, parallel performance, MHD, magnetohydrodynamic

15. NUMBER OF PAGES

12

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
unclassified	unclassified	unclassified	unclassified/unlimited

Parallel Workstation Cluster for Algorithm Development, Parameter Scoping, and Data Analysis

Contract Number: F49620-98-1-0244

U. Shumlak
Department of Aeronautics and Astronautics, Box 352250
University of Washington, Seattle, WA 98195-2250

1 Executive Summary

This final technical report describes the DURIP grant to establish a cluster of workstations which are configured like a parallel computer to test and debug parallel algorithms, to perform preliminary parameter scoping studies, and to post-process data.

The cluster of workstations are connected with a fast network and used as a local parallel computer. This approach has the advantage that it mimics the interprocessor communication and scalability of the large machines at a fraction of the cost. This system uses off-the-shelf hardware and public domain software which makes the system simple to assemble and maintain. Additionally, one of the workstations is a fully capable machine that can be used for post-processing and graphics rendering of data generated on the large machines at a shared resource center.

The parallel workstation cluster has been purchased, assembled, and significantly contributed to our AFOSR project. The parallel workstation cluster accelerates the algorithm development process because it eliminates the delays associated with remote computing on shared resources such as the parallel supercomputers at the Department of Defense High Performance Computing Centers. Algorithms can be debugged locally without encumbering the large machines and without the transfer delays inherent to computing over the internet. The parallel cluster also is used in a dedicated fashion to perform timing comparisons to evaluate algorithms.

The parallel workstation cluster performs simulations to scope a parameter space for optimization and design. A parallel supercomputer is then used for the detailed scoping using fewer runs. The parallel cluster is used for data analysis and graphics rendering of the data from the supercomputer runs as well as the local runs. This approach has helped to alleviate the over-subscription of the shared resources and has allowed us to obtain the computational results in less time.

We are currently funded by the Air Force Office of Scientific Research to develop an algorithm for parallel computers to model the physics of plasmas and to apply the code to support Air Force relevant projects and devices.[1] A local parallel computer has enhanced the quality of our research by allowing us to examine a broader spectrum of solvers for our algorithm. In addition it will allow us to provide better and faster computational support for Air Force projects like the High Power Microwave and the Portable Pulsed Power Programs at the Air Force Phillips Laboratory and the Plasma-based Hypersonic Drive Initiative. All of these programs are explicitly mentioned in the *New World Vistas* Report from the USAF Scientific Advisory Board.[1]

2 Introduction

The last decade has seen great advances in computing power. In large part this is due to the parallel computer which offers the ability scale up small computers to achieve the compute speeds of supercomputers. However, parallel computing has significantly complicated the development of suitable computer codes that can exploit this power. In particular, algorithms that have been developed for serial or vector computers are usually inadequate for parallel computers because they require large amounts of interprocessor communication. Interprocessor communication can cripple the performance of a parallel computer. The need to develop new algorithms that are well suited for parallel computers spawned this effort at the University of Washington.

The Department of Defense has acquired several parallel supercomputers that are located at shared resource centers across the nation. These computers are excellent for production runs by existing codes where jobs are submitted to a batch queueing system and the code output is retrieved when the run is completed. However, these remote supercomputers can be cumbersome for code and algorithm development. Because of the volume of traffic on the internet and the over-subscription of the supercomputers,

[1]Grant No.: F49620-99-1-0084. Program Manager: Dr. Marc Jacobs, 703-696-8409.

answering a single code development decision can take as long as performing a production run.

This type of development paradigm is similar to the one that was in place during the early 1980's with vector computing. Production runs and code development competed for run time on the vector supercomputers. This problem was largely alleviated by local workstations that were fully compatible with the vector supercomputers. Code development could then take place locally on a workstation without any of the delays associated with remote, shared-environment computing, while the vector supercomputers were only used for large production runs.

We believe an analogous strategy would be useful for parallel computers. A small local parallel computer that has the same functionality as the parallel supercomputers could be used for code and algorithm development. The parallel supercomputers could then be reserved for large production runs. The local parallel computer can be composed of off-the-shelf workstations and connected with a fast network. This approach will mimic the interprocessor communication that is used by the parallel supercomputers and still keep costs down. Additionally, the local parallel computer can be used to perform small production runs and to analyze the data from a parallel supercomputer run. This strategy will result in faster algorithm and code development and less burdened supercomputers at the shared resource centers which will result in greater productivity and faster turn-around for Department of Defense applications.

3 Instrument Description

We purchased a cluster of 17 DEC AlphaStation 433 workstations. Sixteen of the workstations are stripped-down compute nodes without any peripherals. Each of these workstations has a local disk for swap space and 512 megabytes of memory. The remaining DEC workstation is a fully capable workstation that is used as the master processor and interfaces to the remaining compute nodes. It performs any extra duties such as spawning tasks on the other nodes, input, and output, when the parallel cluster is used in a master-slave paradigm. The master processor is also used for post-processing data that has been generated either locally or on the large supercomputers at the shared resource centers. The proposal listed that 8 DEC AlphaStation 500/500 workstations would be used to form the cluster. However, DEC was running a special on a slightly slower workstation model with a substantial cost savings. The cost special allowed us to purchase 17 DEC AlphaStation

	Item	Associated Cost
a)	DEC AlphaStation 433 3D UNIX (Graphics Workstation)	$9,174
b)	(16) DEC AlphaStation 433 (Compute Nodes)	$89,144
c)	ConnectPro KVM switch	$2,344
d)	Allied Telesyn 24-port 10/100 Ethernet switch	$1,955
e)	(2) Tektronix XP417c (X-Terminals)	$3,444
f)	NEC Monitor	$542

Table 1: Equipment List

433 workstations for the same price.

The workstations of the parallel cluster are mounted on warehouse shelving in a climate controlled room. A single NEC monitor, keyboard, and mouse are connected to the cluster through a workstation selectable ConnectPro KVM switch. The purchased equipment and the associated cost are listed in Table 1.

We have also ordered 14 dual-processor Intel workstations which will have a local disk for swap space and 256 megabytes of memory. The Intel workstations are symmetric multiprocessor (SMP) machines with shared memory. These workstations emulate the shared memory parallel supercomputers such as the Cray J90 series and the Silicon Graphics Power Challenge and Origin 2000 at the Aeronautical Systems Center (ASC) at Wright-Patterson Air Force Base. The 14 Intel Dual Pentium III workstations are being supplied by the Intel Corporation through its Technology for Education 2000 Program. The 14 Dual Pentium workstations are being ordered instead of the 6 Quad Pentium units because of their higher performance.

Both the DEC and Intel workstations will be connected using a fast ethernet network and a network switch that makes the all nodes appear equi-distant. The network switch is an Allied Telesyn 24-port 10/100 Ethernet managed switch. This distributed processor/memory configuration is the same topology that is implemented in the IBM SP2 at the Maui High Performance Computing Center. Two Tektronix X-terminals are used to interface with the parallel workstation cluster.

The DEC AlphaStations came with the DEC software development tools which include all the necessary compilers and operating system. The workstation cluster use TotalView for parallel debugging needs. We have installed the Message Passing Interface (MPI)[2] and the Parallel Virtual Machine (PVM)[3] communication libraries for internodal communication. MPI and PVM have become the industry standard and are frequently used on the parallel supercomputers at the shared resource centers. These libraries are public-domain software and are available free of charge.

4 Research Project Summary

The primary objective of the AFOSR-funded research project which was supported by parallel workstation cluster is to develop an advanced algorithm for parallel supercomputers to model time-dependent and steady state magnetohydrodynamics (MHD) in all three dimensions.[4] The title of the research project is "An Implicit, Conservative Multi-Temperature MHD Algorithm." A viable time-dependent, three-dimensional MHD code will provide a valuable tool for the design and testing of plasma related technologies that are important to the Air Force and industry. These applications include portable pulsed power, high power microwave devices, advanced plasma thrusters for space propulsion, hypersonic drag reduction, nuclear weapons effects simulations, radiation production for counter proliferation, and fusion for power generation. Implementing the algorithm on parallel supercomputers allows the detailed modeling of realistic plasmas in complex three-dimensional geometries.

We have developed a time-dependent, three-dimensional, arbitrary-geometry MHD algorithm with viscous and resistive effects and tested the code against known analytical problems. We have implemented the algorithm on a parallel architecture and optimized the parallelization strategy. The algorithm has been cast using unaligned finite volumes, instead of generalized coordinates, which has greatly improved the accuracy of the code. Global second-order accuracy was achieved by using second-order boundary conditions for both internal (interblock) and external (physical) boundaries. Future plans include investigating more powerful implicit solvers, extending the code to model multi-temperature effects including the presence of neutral gas molecules.

Plasmas are essential to many technologies that are important to the Air Force, many of which have dual-use potential. These applications include portable pulsed power, high power microwave devices, hypersonic drag re-

duction, advanced plasma thrusters for space propulsion, nuclear weapons effects simulations, radiation production for counter proliferation, and fusion for power generation. In general, plasmas fall into a density regime where they exhibit both collective (fluid) behavior and individual (particle) behavior. Many plasmas of interest can be modeled by treating the plasma like a conducting fluid and assigning macroscopic parameters that accurately describe its particle-like interactions. The magnetohydrodynamic (MHD) model is a plasma model of this type.

The three-dimensional, viscous, resistive MHD plasma model is a set of mixed hyperbolic and parabolic equations. The Navier-Stokes equations are also of this type. This project applies some advances that have been made in implicit algorithms for the Navier-Stokes equations to the MHD equations. These implicit algorithms solve the equation set in a fully coupled manner, which generates better accuracy than the current methods used for MHD simulations.

When expressed in conservative, non-dimensional form, the MHD model is described by the following equation set.

$$\frac{\partial}{\partial t}\begin{bmatrix} \rho \\ \rho\mathbf{v} \\ \mathbf{B} \\ e \end{bmatrix} + \nabla\cdot\begin{bmatrix} \rho\mathbf{v} \\ \rho\mathbf{v}\mathbf{v} - \mathbf{B}\mathbf{B} + (p + \mathbf{B}\cdot\mathbf{B}/2)\bar{\bar{\mathrm{I}}} \\ \mathbf{v}\mathbf{B} - \mathbf{B}\mathbf{v} \\ (e + p + \mathbf{B}\cdot\mathbf{B}/2)\,\mathbf{v} - (\mathbf{B}\cdot\mathbf{v})\,\mathbf{B} \end{bmatrix} =$$

$$\nabla\cdot\begin{bmatrix} 0 \\ (ReAl)^{-1}\,\bar{\bar{\tau}} \\ (RmAl)^{-1}\,\bar{\bar{\Xi}}(\bar{\bar{\eta}},\mathbf{B}) \\ (ReAl)^{-1}\,\mathbf{v}\cdot\bar{\bar{\tau}} - (RmAl)^{-1}\,\bar{\bar{\eta}}\cdot(\nabla\times\mathbf{B})\times\mathbf{B} + \frac{M_i}{2}\,(PeAl)^{-1}\,\bar{\bar{k}}\cdot\nabla T \end{bmatrix} \quad (1)$$

The variables are density (ρ), velocity ($\mathbf{v}$), magnetic induction ($\mathbf{B}$), pressure (p), energy density (e), and temperature (T). $\bar{\bar{\Xi}}(\bar{\bar{\eta}},\mathbf{B})$ is the transverse resistive electric field tensor. M_i is the ion mass. The energy density is

$$e = \frac{p}{\gamma - 1} + \rho\frac{\mathbf{v}\cdot\mathbf{v}}{2} + \frac{\mathbf{B}\cdot\mathbf{B}}{2} \quad (2)$$

where $\gamma = c_p/c_v$ is the ratio of the specific heats. The non-dimensional tensors are the stress tensor ($\bar{\bar{\tau}}$), the electrical resistivity ($\bar{\bar{\eta}}$), and the thermal conductivity ($\bar{\bar{k}}$), and $\bar{\bar{\mathrm{I}}}$ is the identity matrix. The non-dimensional numbers

are defined as follows:

$$\begin{array}{lll} \text{Alfvén Number :} & Al & \equiv V_A/V \\ \text{Reynolds Number :} & Re & \equiv LV/\nu \\ \text{Magnetic Reynolds Number :} & Rm & \equiv \mu_o LV/\eta \\ \text{Péclet Number :} & Pe & \equiv LV/\kappa \end{array} \tag{3}$$

The characteristic variables are length (L), velocity (V), Alfvén speed ($V_A = B/\sqrt{\mu_o \rho}$), kinematic viscosity ($\nu$), electrical resistivity (η), and thermal diffusivity ($\kappa = k/\rho c_p$). μ_o is the permeability of free space ($4\pi \times 10^{-7}$).

For convenience, the MHD equation set [eqn(1)] is rewritten in the following compact form

$$\frac{\partial Q}{\partial t} + \nabla \cdot \bar{\bar{T}}_h = \nabla \cdot \bar{\bar{T}}_p, \tag{4}$$

where Q is the vector of conservative variables, $\bar{\bar{T}}_h$ is the tensor of hyperbolic fluxes, and $\bar{\bar{T}}_p$ is the tensor of parabolic fluxes. The forms of these vectors and tensors can be seen from eqn(1). The hyperbolic fluxes are associated with wave-like motion, and the parabolic fluxes are associated with diffusion-like motion.

We have applied our code to study the nonlinear phase of the tilt instability in compact tori. Compact tori are currently being experimentally investigated at the Air Force Research Laboratory. The parallel performance of the workstation cluster is shown in Figure 1 and Figure 2. Figure 1 is a fixed problem size that is further divided as the number of processors is increased. The theoretical speedup is equal to the number of processors. A more significant parallel performance test is to hold the problem size per processor constant. The simulation then scales with the number of processors and the theoretical speedup is unity. These results are shown in Figure 2. For comparison the same simulation was performed on the IBM SP2 at the Maui High Performance Computing Center. The SP2 results are shown in Figure 3. Almost identical parallel performance is attained on both the IBM SP2 and our local parallel workstation cluster. However, the primary difference is that we were able to obtain our local results in about an hour while the wait in the batch que of the SP2 was 2 days. This enables us to debug our MHD code in much less time than relying on the SP2 for debugging. Furthermore, it preserves the SP2 for production runs that are better suited to take advantage of its massively parallel scalability.

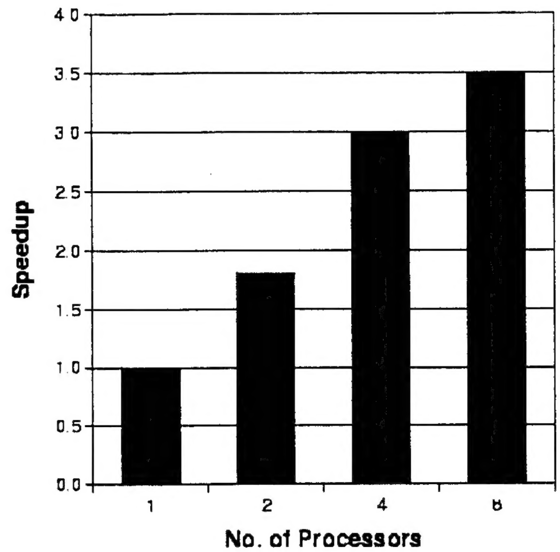

Figure 1: Fixed grid speedup results for the parallel workstation cluster. The simulation is a 3-D compact toroid tilt instability.

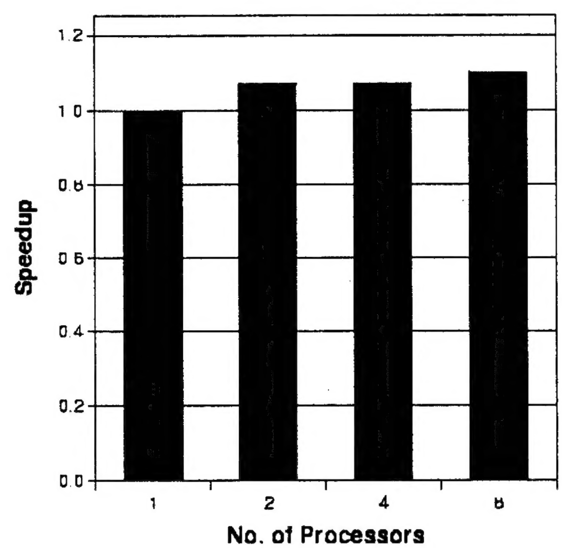

Figure 2: Scaled grid speedup results for the parallel workstation cluster. The simulation is a 3-D compact toroid tilt instability.

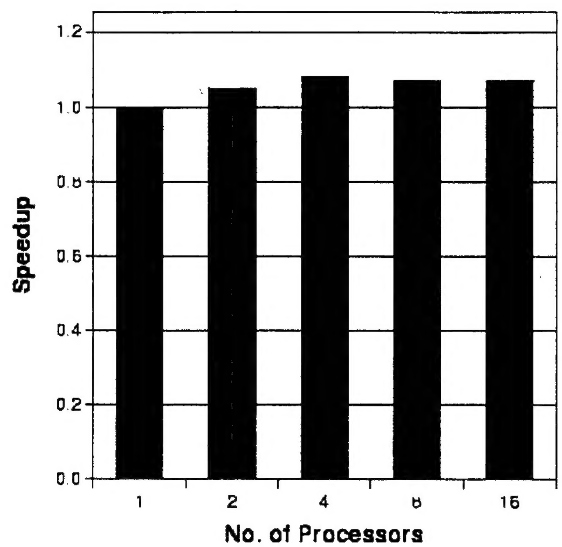

Figure 3: Scaled grid speedup results for the IBM SP2 at the Maui High Performance Computing Center. The simulation is a 3-D compact toroid tilt instability.

References

[1] G. H. McCall, J. A. Corder and the USAF Scientific Advisory Board, *New World Vistas, Air and Space Power for the 21st Century,* United States Air Force Document, Summary Volume (December 1995).

[2] Message Passing Interface (MPI) standard is a new library specification for message-passing, proposed as a standard by a broadly based committee of vendors, implementors, and users. MPI is supported by ARPA, NSF, and the Commission of the European Community, University of Tennessee, Knoxville.

[3] Al Geist, A. Beguelin, J. Dongara, W. Jiang, R. Manchek, V. Sunderam, *PVM 3 User's Guide and Reference Manual,* 1994.

[4] O. S. Jones, U. Shumlak, D. S. Eberhardt, J. Comput. Phys. **130**, 231 (1997).